PRACTICAL-5

AIM:

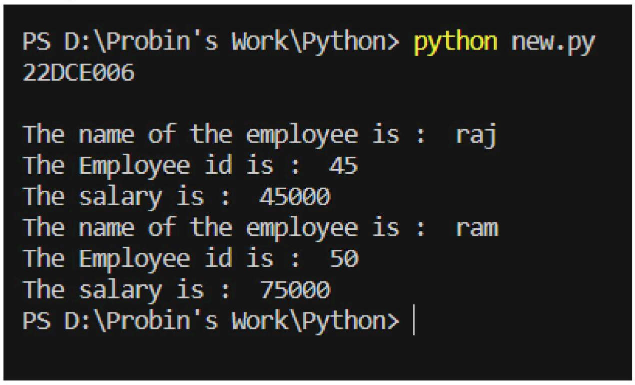
A) Create a class employee and display employee details
B) From above create class and count number of employee and display a salary
amount, if the salary is raised to 1.04%.
C) Fetch children class details using different types of inheritance (Single, Multilevel,
and Multiple) With constructor
D) Find who will be first among two students using polymorphism.

A)

Program Code:

```
print("22DCE006\n")
class emp:
    c=0
    def __init__(self,name,e_id,salary):
        self.name=name
        self.e_id=e_id
        self.salary=salary
        emp.c+=1
    def disp(self):
        print("The name of the employee is : ",self.name)
        print("The Employee id is : ",self.e_id)
        print("The salary is : ",self.salary)
p1=emp("raj",45,45000)
p1.disp()
p2=emp("ram",50,75000)
p2.disp()
```

Output:



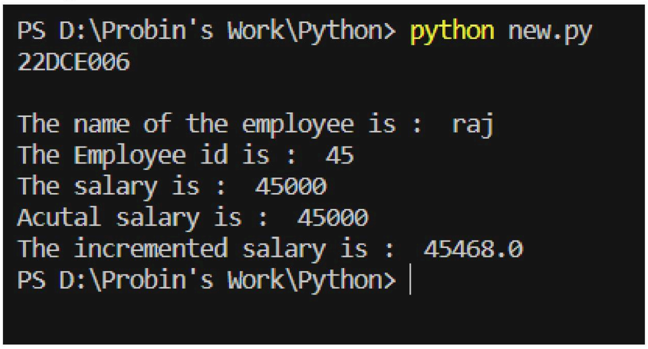
```
PS D:\Probin's Work\Python> python new.py
22DCE006

The name of the employee is :  raj
The Employee id is :  45
The salary is :  45000
The name of the employee is :  ram
The Employee id is :  50
The salary is :  75000
PS D:\Probin's Work\Python>
```

B)

Program Code:

```
print("22DCE006\n")
class emp:
    c=0
    def __init__(self,name,e_id,salary):
        self.name=name
        self.e_id=e_id
        self.salary=salary
        emp.c+=1
    def disp(self):
        print("The name of the employee is : ",self.name)
        print("The Employee id is : ",self.e_id)
        print("The salary is : ",self.salary)
    def inc(self):
        inc=(self.salary*0.0104)+self.salary
        print("Acutal salary is : ",self.salary)
        print("The incremented salary is : ",inc)

p1=emp("raj",45,45000)
p1.disp()
p1.inc()
```

Output:



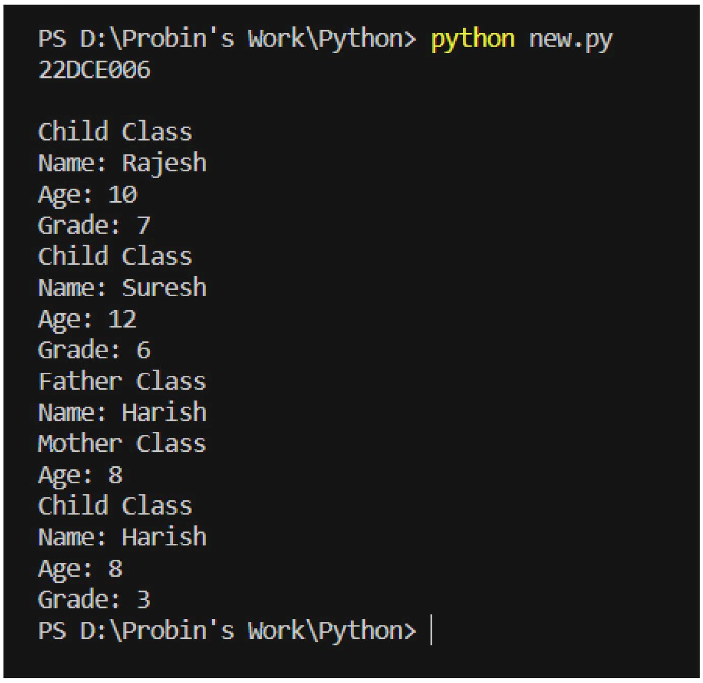
```
PS D:\Probin's Work\Python> python new.py
22DCE006

The name of the employee is :  raj
The Employee id is :  45
The salary is :  45000
Acutal salary is :  45000
The incremented salary is :  45468.0
PS D:\Probin's Work\Python> 
```

C)

Program Code:

```
print("22DCE006\n")
# Single Inheritance
class Parent:
    def __init__(self, name):
        self.name = name

    def display(self):
        print("Parent Class")

class Child(Parent):
    def __init__(self, name, age):
        super().__init__(name)
        self.age = age

    def display_child(self):
        print("Child Class")
        print(f"Name: {self.name}")
        print(f"Age: {self.age}")

# Multilevel Inheritance
class GrandParent:
    def __init__(self, name):
        self.name = name

    def display(self):
        print("GrandParent Class")

class Parent(GrandParent):
    def __init__(self, name, age):
        super().__init__(name)
        self.age = age

    def display_parent(self):
        print("Parent Class")
        print(f"Name: {self.name}")
        print(f"Age: {self.age}")

class Child(Parent):
    def __init__(self, name, age, grade):
        super().__init__(name, age)
        self.grade = grade

    def display_child(self):
        print("Child Class")
        print(f"Name: {self.name}")
        print(f"Age: {self.age}")
        print(f"Grade: {self.grade}")

# Multiple Inheritance
class Father:
    def __init__(self, name):
        self.name = name

    def display_father(self):
        print("Father Class")
        print(f"Name: {self.name}")

class Mother:
    def __init__(self, age):
        self.age = age

    def display_mother(self):
        print("Mother Class")
        print(f"Age: {self.age}")

class Child(Father, Mother):
    def __init__(self, name, age, grade):
        super().__init__(name)
        Mother.__init__(self, age)
        self.grade = grade

    def display_child(self):
        print("Child Class")
        print(f"Name: {self.name}")
        print(f"Age: {self.age}")
        print(f"Grade: {self.grade}")

# Single Inheritance Example
child1 = Child("Rajesh", 10 , 7)
child1.display_child()

# Multilevel Inheritance Example
child2 = Child("Suresh", 12, 6)
child2.display_child()

# Multiple Inheritance Example
child3 = Child("Harish", 8, 3)
child3.display_father()
child3.display_mother()
child3.display_child()
```

Output:



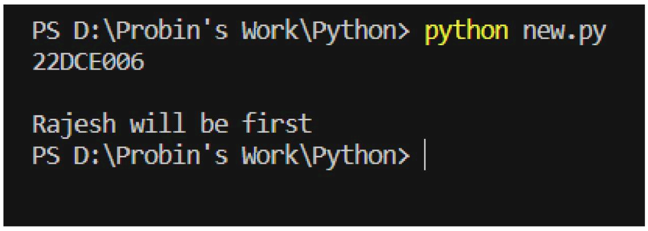
```
PS D:\Probin's Work\Python> python new.py
22DCE006

Child Class
Name: Rajesh
Age: 10
Grade: 7
Child Class
Name: Suresh
Age: 12
Grade: 6
Father Class
Name: Harish
Mother Class
Age: 8
Child Class
Name: Harish
Age: 8
Grade: 3
PS D:\Probin's Work\Python> 
```

D)

Program Code:

```
print("22DCE006\n")
class Student:
   def __init__(self, name, id):
      self.name = name
      self.id = id

   def __lt__(self, other):
      return self.id < other.id

student1 = Student("Rajesh", 1)
student2 = Student("Neeraj", 2)

if student1 < student2:
   print(f"{student1.name} will be first")
else:
   print(f"{student2.name} will be first")
```

Output:



```
PS D:\Probin's Work\Python> python new.py
22DCE006

Rajesh will be first
PS D:\Probin's Work\Python> 
```

Conclusion: From this practical we learned the concept of class and constructors. Also learned about the different types of inheritance , its usage and polymorphism.

Signature:

Grade:

Remarks by the Staff: